

Защита в операционных системах

Вязанкин Олег Николаевич

к.ф.-м.н., доцент кафедры ПОЗИ, ИМИТ

ovyazankin@gmail.com

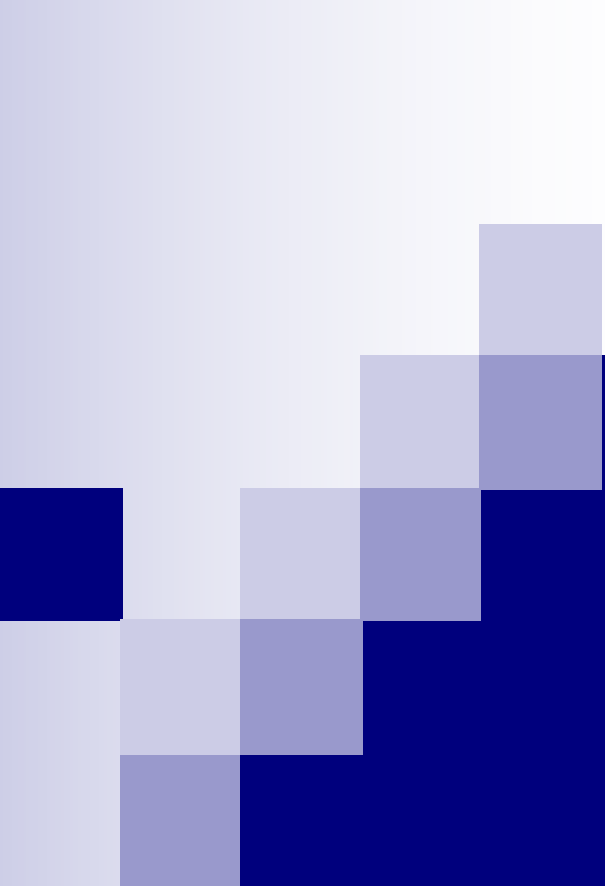
Литература

■ Основная

- Проскурин В.Г. Защита в операционных системах. – М.: Горячая линия – Телеком, 2014.
- Безбогов А.А., Яковлев А.В., Мартемьянов Ю.Ф. Безопасность операционных систем. – М.: Машиностроение, 2007.
- Хорев П.Б. Программно-аппаратная защита информации. – М.: Форум, 2012.

■ Дополнительная

- Таненбаум Э., Х. Бос Современные операционные системы. - СПб.: Питер, 2015.
- Столлингс В. Операционные системы. - М.: Вильямс, 2002.
- Робачевский А.М., Немнюгин С.А., Стесик О.Л. Операционная система UNIX. – СПб.: БХВ – Петербург, 2008.



Тема 1. Понятие защищенной ОС. Основные встроенные механизмы защиты ОС.



План

- Основные определения
- Основные подходы к построению защищенных ОС
- Идентификация и аутентификация
- Разграничение прав доступа к объектам ОС
- Выявление вторжений. Аудит системы защиты.

Основные определения

- *ОС будем называть защищенной*, если она предусматривает средства защиты от основных угроз конфиденциальности, целостности и доступности информации, актуализированных с учетом особенностей эксплуатации данного конкретного экземпляра ОС.
- *Политикой безопасности* будем называть набор норм, правил и практических приемов, регламентирующих порядок хранения и обработки ценной информации.
- *Адекватной политикой безопасности* будем называть такую политику безопасности, которая обеспечивает достаточный уровень защищенности ОС.

Важно! При адекватной политике не обязательно достигается максимально возможная защищенность системы.

Основные определения

- *Конфиденциальность (confidentiality)* - гарантия того, что секретные данные будут доступны только тем пользователям, которым этот доступ разрешен (такие пользователи называются авторизованными).
- *Доступность (availability)* - гарантия того, что авторизованные пользователи всегда получают доступ к данным.
- *Целостность (integrity)* - гарантия сохранности данными правильных значений, которая обеспечивается запретом для неавторизованных пользователей каким-либо образом изменять, модифицировать, разрушать или создавать данные.

Об адекватной политике безопасности (АПБ)

- Задача выбора и поддержания АПБ является важной и сложной задачей, стоящей перед сисадмином.
- Не всякая АПБ применима на практике. Чем лучше ОС защищена, тем труднее с ней работать пользователям и администратором. Причины следующие:
 - система защиты не обладает интеллектом;
 - больше защитных функций – больше времени и средств на поддержание защиты;
 - подсистема защиты ОС потребляет аппаратные ресурсы компьютера;
 - поддержание слишком жесткой политики безопасности может негативно сказаться на надежности функционирования ОС.

Таким образом, при определении АПБ не следует пытаться достигнуть максимально возможного уровня защищенности ОС. Оптимальная АПБ – такая политика безопасности, которая не только не позволяет нарушителям выполнять несанкционированные действия, но и не приводит к вышеуказанным негативным последствиям.



Основные подходы к построению защищенных ОС

Существует два основных подхода к созданию защищенных ОС.

- Фрагментарный
- Комплексный

Основные подходы к построению защищенных ОС

■ Фрагментарный подход

При фрагментарном подходе вначале организуется защита от одной угрозы, затем от другой и т.д. Примером фрагментарного подхода может служить ситуация, когда за основу берется незащищенная операционная система (например, Windows 95), на нее устанавливаются антивирусный пакет, система шифрования, система регистрации действий пользователей и т.д.

Основной недостаток фрагментарного подхода очевиден - при применении этого подхода подсистема защиты операционной системы представляет собой набор разрозненных программных продуктов, как правило, произведенных разными производителями. Эти программные средства работают независимо друг от друга, организовать их тесное взаимодействие практически невозможно. Кроме того, отдельные элементы такой подсистемы защиты могут некорректно работать в присутствии друг друга, что приводит к резкому снижению надежности системы.



Основные подходы к построению защищенных ОС

■ Комплексный подход

При комплексном подходе к организации защиты системы защитные функции вносятся в операционную систему на этапе проектирования архитектуры операционной системы и являются ее неотъемлемой частью. Отдельные элементы подсистемы защиты, созданной на основе комплексного подхода, тесно взаимодействуют друг с другом при решении различных задач, связанных с организацией защиты информации. Поскольку подсистема защиты разрабатывается и тестируется в совокупности, конфликты между ее отдельными компонентами практически невозможны.



Идентификация и аутентификация

Для начала рассмотрим проблему контроля доступа в систему. Наиболее распространенным способом контроля доступа является процедура регистрации. Обычно каждый пользователь в системе имеет уникальный идентификатор. Идентификаторы пользователей применяются с той же целью, что и идентификаторы любых других объектов, файлов, процессов. Идентификация заключается в сообщении пользователем своего идентификатора. Для того чтобы установить, что пользователь именно тот, за кого себя выдает, то есть что именно ему принадлежит введенный идентификатор, в информационных системах предусмотрена процедура аутентификации (authentication, опознавание, в переводе с латинского означает "установление подлинности"), задача которой - предотвращение доступа к системе нежелательных лиц.



Идентификация и аутентификация

Обычно аутентификация базируется на одном или более из трех пунктов:

- то, чем пользователь владеет (ключ или магнитная карта);
- то, что пользователь знает (пароль);
- атрибуты пользователя (отпечатки пальцев, подпись, голос).



Идентификация и аутентификация

Пароли и их уязвимость

Недостатки паролей связаны с тем, что трудно сохранить баланс между удобством пароля для пользователя и его надежностью. Пароли могут быть угаданы, случайно показаны или нелегально переданы авторизованным пользователем неавторизованному.

Существует два общих способа угадать пароль:

- сбор информации о пользователе;
- перебор всех наиболее вероятных комбинаций букв, чисел и прочих знаков (атака по словарю).

Чтобы заставить пользователя выбрать трудно угадываемый пароль, во многих системах внедрена реактивная проверка паролей, которая при помощи собственной программы-взломщика паролей может оценить качество пароля, введенного пользователем.

Идентификация и аутентификация

Шифрование пароля

Для хранения секретного списка паролей на диске во многих ОС используется криптография. Система задействует одностороннюю функцию, которую просто вычислить, но для которой чрезвычайно трудно (разработчики надеются, что невозможно) подобрать обратную функцию.

Примеры:

- UNIX (модифицированный вариант алгоритма DES)
- Windows NT (DES + MD4)

При удаленном доступе к ОС нежелательна передача пароля по сети в открытом виде. Одним из типовых решений является использование криптографических протоколов. В качестве примера можно рассмотреть протокол опознавания с подтверждением установления связи путем вызова – CHAP (Challenge Handshake Authentication Protocol).

В системах, работающих с большим количеством пользователей, когда хранение всех паролей затруднительно, применяются для опознавания сертификаты, выданные доверенной стороной.

Авторизация. Разграничение прав доступа к объектам ОС

■ Авторизация

- После успешной регистрации система должна осуществлять авторизацию (authorization) – предоставление субъекту (процесс, пользователь) прав на доступ к объекту (процесс, память, принтер, файл, семафор и т.д.). Средства авторизации контролируют доступ легальных пользователей к ресурсам системы, предоставляя каждому из них именно те права, которые были определены администратором, а также осуществляют контроль возможности выполнения пользователем различных системных функций.

Авторизация. Разграничение прав доступа к объектам ОС

■ Правила разграничения доступа

- Компьютерная система может быть смоделирована как набор субъектов и объектов.
- Каждый объект имеет уникальное имя, отличающее его от других объектов в системе, и каждый из них может быть доступен через хорошо определенные и значимые операции.
- Операции зависят от объектов.
- Желательно добиться того, чтобы процесс осуществлял авторизованный доступ только к тем ресурсам, которые ему нужны для выполнения его задачи. Это требование минимума привилегий, полезно с точки зрения ограничения количества повреждений, которые процесс может нанести системе.

Различают дискреционный (избирательный) способ управления доступом и полномочный (мандатный).

Авторизация. Разграничение прав доступа к объектам ОС

Мандатное управление доступом

Данная модель управления доступом в основном предназначена для предотвращения утечки конфиденциальной информации.

Данный подход заключается в том, что все объекты могут иметь уровни секретности (*мандатные метки*), а все субъекты делятся на группы, образующие иерархию в соответствии с уровнем допуска к информации.

При каждой передаче информации от объекта к объекту вместе с информацией передается мандатная метка. Если, например, пользователь попытается скопировать информацию из файла с мандатной меткой «секретно» в файл с мандатной меткой «несекретно», мандатная метка файла-приемника будет изменена на «секретно» либо операция копирования будет запрещена.

Авторизация. Разграничение прав доступа к объектам ОС

Мандатное управление доступом

Иногда эту модель называют моделью многоуровневой безопасности, которая должна обеспечивать выполнение следующих правил.

- Простое свойство секретности. Субъект может читать информацию только из объекта, уровень секретности которого не выше уровня секретности субъекта (например, генерал читает документы лейтенанта, но не наоборот). Это так называемое правило *NRU* (not read up – не читать выше).
- Субъект может записывать информацию в объекты только своего уровня или более высоких уровней секретности (например, генерал не может случайно разгласить нижним чинам секретную информацию). Это так называемое правило *NWD* (not write down – не записывать ниже).

Авторизация. Разграничение прав доступа к объектам ОС

Мандатное управление доступом

Данная модель управления доступом имеет ряд существенных недостатков.

- Существенно страдает производительность системы (права доступа к объекту проверяются перед каждой операцией над ним).
- Создает серьезные проблемы, связанные с тем, что если уровень конфиденциальности процесса строго выше нуля, то вся информация в памяти процесса фактически является секретной и не может быть записана в несекретный объект.
- Вызывает проблемы вопрос о назначении грифов секретности вновь создаваемым объектам (процесс с ненулевым уровнем секретности не может создать объект с грифом секретности ниже уровня конфиденциальности процесса, что во многих ситуациях неудобно).
- Наконец, данная модель разработана для хранения секретов, но не гарантирует целостности данных. (например, здесь лейтенант имеет право писать в файлы генерала.)

Авторизация. Разграничение прав доступа к объектам ОС

Дискреционное управление доступом

При дискреционном доступе, определенные операции над конкретным объектом запрещаются или разрешаются субъектам или группам субъектов. Система правил для дискреционного управления доступом формулируется следующим образом:

- Для любого объекта системы существует владелец.
- Владелец объекта может произвольно ограничивать доступ других субъектов к данному объекту.
- Для каждой тройки субъект-объект-право возможность доступа определена однозначно.
- Существует хотя бы один привилегированный пользователь, имеющий возможность обратиться к любому объекту по любому методу доступа.

Авторизация. Разграничение прав доступа к объектам ОС

Дискреционное управление доступом

Для определения прав доступа субъектов к объектам используется *матрица доступа*. Строки – объекты, столбцы – субъекты (или наоборот). В каждой ячейке матрицы доступа хранится совокупность прав доступа, предоставленных данному субъекту на данный объект.

Для сокращения объема матрицы доступа используется объединение субъектов доступа в группы. Права, предоставленные группе субъектов, автоматически предоставляются каждому субъекту группы.

Вместе с каждым объектом доступа хранятся его *атрибуты защиты*, описывающие, кто является владельцем объекта и каковы права доступа к данному объекту различных субъектов. Атрибуты защиты фактически представляют собой совокупность идентификатора владельца и строки матрицы доступа в кодированном виде.

Авторизация. Разграничение прав доступа к объектам ОС

Дискреционное управление доступом

При таком способе реализации управления доступов имеет место проблема, связанная с тем, что значения элементов матрицы доступа могут противоречить друг другу. Для решения проблемы правила управления доступом должны включать в себя правила разрешения подобных противоречий.

При создании нового объекта владелец объекта должен определить права доступа различных субъектов к этому объекту. Если владелец объекта этого не сделал, новому объекту либо назначаются атрибуты защиты по умолчанию, либо новый объект наследует атрибуты защиты от объекта-контейнера, в котором создается объект.

Большинство операционных систем реализуют именно дискреционное управление доступом. Главные его достоинства – простота реализации и гибкость, основные недостатки – рассредоточенность управления и сложность централизованного контроля. Вместе с тем, общая защищенность компьютерной системы на основе только лишь дискреционного управления доступом во многих случаях недостаточна.

Авторизация. Разграничение прав доступа к объектам ОС

Изолированная программная среда (изучить самостоятельно)

Каждая из приведенных моделей разграничения доступа имеет свои достоинства и недостатки. Приведенная таблица позволяет провести их сравнительный анализ.

	Управление доступом		
	избирательное	изолированная среда	полномочное
Защита от утечки информации	отсутствует	отсутствует	имеется
Защищенность от прог. закладок	низкая	высокая	низкая
Сложность реализации	низкая	средняя	высокая
Сложность администрирования	низкая	средняя	высокая
Затраты ресурсов компьютера	низкие	низкие	высокие
Использование стороннего ПО	возможно	проблематично	проблематично



Выявление вторжений. Аудит системы защиты

Даже самая лучшая система защиты рано или поздно будет взломана. Обнаружение попыток вторжения является важнейшей задачей системы защиты, поскольку ее решение позволяет минимизировать ущерб от взлома и собирать информацию о методах вторжения. Как правило, поведение взломщика отличается от поведения легального пользователя. Иногда эти различия можно выразить количественно, например подсчитывая число некорректных вводов пароля во время регистрации.

Основным инструментом выявления вторжений является запись данных аудита. Отдельные действия пользователей протоколируются, а полученный протокол используется для выявления вторжений.

Выявление вторжений. Аудит системы защиты

Аудит, таким образом, заключается в регистрации специальных данных о различных типах событий, происходящих в системе и так или иначе влияющих на состояние безопасности компьютерной системы. К числу таких событий обычно причисляют следующие:

- вход или выход из системы;
- операции с файлами (открыть, закрыть, переименовать, удалить);
- обращение к удаленной системе;
- смена привилегий или иных атрибутов безопасности (режима доступа, уровня благонадежности пользователя и т. п.).

Если фиксировать все события, объем регистрационной информации, скорее всего, будет расти слишком быстро, а ее эффективный анализ станет невозможным. Следует предусматривать наличие средств выборочного протоколирования как в отношении пользователей, когда слежение осуществляется только за подозрительными личностями, так и в отношении событий. Слежка важна в первую очередь как профилактическое средство. Можно надеяться, что многие воздержатся от нарушений безопасности, зная, что их действия фиксируются.

Выявление вторжений. Аудит системы защиты

Помимо протоколирования, можно периодически сканировать систему на наличие слабых мест в системе безопасности. Такое сканирование может проверить разнообразные аспекты системы:

- короткие или легкие пароли;
- неавторизованные set-uid программы, если система поддерживает этот механизм;
- неавторизованные программы в системных директориях;
- долго выполняющиеся программы;
- нелогичная защита как пользовательских, так и системных директорий и файлов. Примером нелогичной защиты может быть файл, который запрещено читать его автору, но в который разрешено записывать информацию постороннему пользователю;
- потенциально опасные списки поиска файлов, которые могут привести к запуску "троянского коня";
- изменения в системных программах, обнаруженные при помощи контрольных сумм.

Любая проблема, обнаруженная сканером безопасности, может быть как ликвидирована автоматически, так и передана для решения администратору системы.